



Worksheet 1 Programming Basics **Answers**

Task 1

1. All IDEs have slightly different facilities, but you can get an idea of them by trying out a built-in debugging module in Python.

In this exercise you will try out the TRACE facility, which enables you to step through the program a line at a time and display the value of variables.

Open your IDE and type the following Python program:

```
import pdb
a = 5
b = 10
c = 6
answer1 = a + b * c
print ("answer1 =", answer1)
answer2 = (a + b) * c
print ("answer2 =", answer2)
```

import pdb imports the **Python DeBugging** module. At the moment, the program does not make use of the module.

Run the program. It should print

```
answer1 = 65
answer2 = 90
```

(a) Set a trace, starting just before the line `b = 10`.

Insert the statement `pdb.set_trace()` between lines `a = 5` and `b = 10`

Save and run the program again. When the program encounters the line `pdb.set_trace()`, it will stop, display the current statement (the line that will execute next), and wait for your input.

You will see the `pdb` prompt,

(Pdb)

(b) Execute the next statement by typing `n` at the `pdb` prompt.

You can type `n` repeatedly, and it will execute one line at a time. Or, you can just press Enter, which will repeat the previous command.

(c) Print the value of the variables by typing `p <variable name>`

e.g. `p answer1`

If you have not yet reached the line where `answer1` is defined, you will get an error message.

(d) Turn off the `pdb` prompt by typing `c` (for “continue”)

You can practise these commands on other programs that you write.



Task 2

2. Write an algorithm that will calculate the amount of paint required to paint a room. The user will enter the dimensions of the room, the total dimensions of the unpaintable areas (such as windows, doors or brickwork) and the number of coats of paint required.

Assume that 1 litre of paint covers 11 sq m.

You can get some handy tips from the site below:

<https://www.dulux.co.uk/en/decorating-tips-and-advice/how-to-calculate-the-right-amount-of-paint>

```
input dimensions of room and unpaintable area
input the number of coats of paint required
calculate totalArea = height * (length + width) * 2
input unpaintedSpaceArea
paintableArea = (totalArea - unpaintedSpaceArea)
input number of coats required
totalPaintArea = paintableArea * numCoats
litresNeeded = totalPaintArea / 11
print litresNeeded
```

In more formal pseudocode:

```
height = input ("Enter height of room: ")
length = input ("Enter length of room: ")
width = input ("Enter width of room: ")
totalArea = height * (length + width) * 2
unpaintedSpaceArea = input ("Enter unpainted space area: ")
paintableArea = totalArea - unpaintedSpaceArea
numCoats = input ("Enter number of coats of paint required: ")
totalPaintArea = paintableArea * numCoats
litresNeeded = totalPaintArea/11          #assume 11 m²/L
print (litresNeeded)
```

Task 3

3. Write pseudocode for a program which calculates the number of miles per gallon a car is doing. The user will input
 - the car mileage the last time the car was filled
 - the car mileage now
 - the total number of litres taken to fill the tank

n.b. There are 0.22 gallons in a litre, or 4.546 litres in a gallon

```
litresPerGallon = 4.546
```



```
previousCarMileage = input("Enter previous car mileage: ")
currentCarMileage = input("Enter current car mileage: ")
litresToFillTank = input("Enter litres to fill tank: ")
totalMiles = currentCarMileage - previousCarMileage
gallonsToFillTank = litresToFillTank/litresPerGallon
mileage = totalMiles/gallonsToFillTank
print("Miles per gallon", mileage)
```

Which of the identifiers in your program could you define as

- (i) a constant? (What is the advantage of doing this?)

litresPerGallon. It is easy to read and use the identifier. Defining it as a constant will reduce the chance of accidentally changing the value. (Note that you cannot define a constant in Python.)

- (ii) an integer

lastCarMileage, currentCarMileage as it is not necessary to be accurate to fractions of a mile – though you could use real numbers here.

- (iii) a real (decimal) number?

litresPerGallon, litresToFillTank, gallonsToFillTank, mileage, as these will involve fractions of a litre/gallon.

Task 4

4. Write an algorithm using pseudocode that asks the user to input the number of students and the number of books to be equally divided between them. Calculate and output the number of books that each student will receive and the number left over.

```
numberOfStudents = input("Enter number of students: ")
numberOfBooks = input("Enter number of books: ")
booksPerStudent = numberOfBooks div numberOfStudents
leftOver = numberOfBooks mod numberOfStudents
print("Books per student: ", booksPerStudent)
print("Books left over: ", leftover)
```

5. Write pseudocode for an algorithm that prompts the user to enter a name, uses a string function to find its length and then tells the user how long the name is.

```
name = input("Please enter your name: ")
length = len(name)
print("There are: ", length, " characters in your name")
```

Extension task



Write the program for Task 1, calculating litres of paint required. Practise using the debugging facilities in Python or an alternative language that you are using, or in the IDE itself.

(See Python and VB programs in relevant folders)